Homework 1

Deadline: 2017.03.13 (Tuesday) 23:59

Problem 1: Candy Crush [hw11.py]

This problem is about implementing a basic elimination for Candy Crush. Given a 2D integer array `board` representing the grid of candy, different positive integers `board[i][j]` represent different types of candies. A value of `board[i][j] = 0` represents that the cell at position `(i, j)` is empty. The given board represents the state of the game following the player's move. Now, you need to restore the board to a stable state by crushing candies according to the following rules:

(1) If three or more candies of the same type are **adjacent vertically or horizontally**, **"crush" them all at the same time** - these positions become empty.

(2) After crushing all candies simultaneously, if an empty space on the board has candies on top of itself, then these candies will drop until they hit a candy or bottom at the same time. (No new candies will drop outside the top boundary.)

(3) After the above steps, there may exist more candies that can be crushed. If so, you need to repeat the above steps.

(4) If there does not exist more candies that can be crushed (ie. the board is stable), then return the current board.

You need to perform the above rules until the board becomes stable, then return the current board.

Sample Input/Output: **(read from candy_input.txt, output to candy_output.txt)**

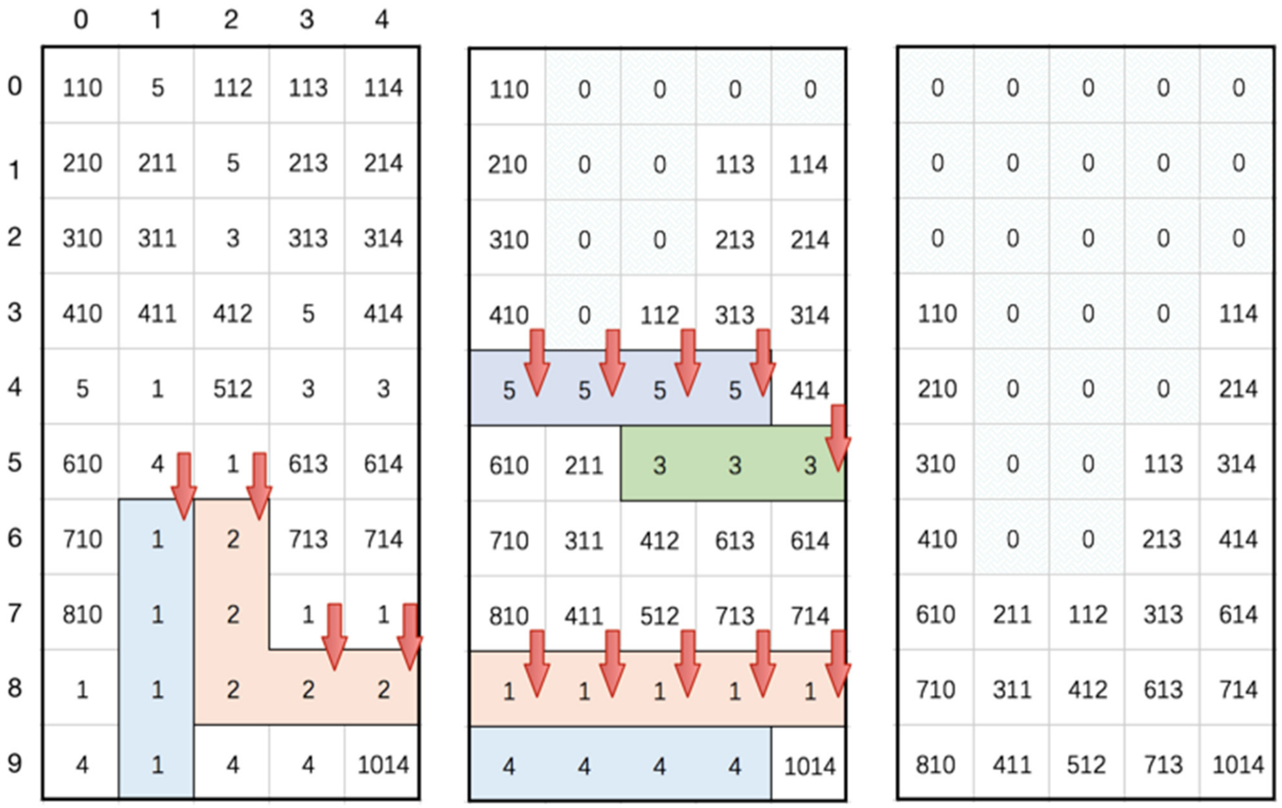
Example Input 1:

[[110,5,112,113,114],[210,211,5,213,214],[310,311,3,313,314],[410,411,412,5,414],[5,1,512,3,3],[610,4,1,613,614],[710,1,2,713,714],[810,1,2,1,1],[1,1,2,2,2],[4,1,4,4,1014]]

Example Output 1:

[[0,0,0,0,0],[0,0,0,0,0],[0,0,0,0,0],[110,0,0,0,114],[210,0,0,0,214],[310,0,0,113,314],[410,0,0,213,414],[610,211,112,313,614],[710,311,412,613,714],[810,411,512,713,1014]]

Step-by-step Example 1:



	0	1	2	3	4
0	110	5	112	113	114
1	210	211	5	213	214
2	310	311	3	313	314
3	410	411	412	5	414
4	5	1	512	3	3
5	610	4	1	613	614
6	710	1	2	713	714
7	810	1	2	1	1
8	1	1	2	2	2
9	4	1	4	4	1014

Candy crush and Drop

110	0	0	0	0
210	0	0	113	114
310	0	0	213	214
410	0	112	313	314
5	5	5	5	414
610	211	3	3	3
710	311	412	613	614
810	411	512	713	714
1	1	1	1	1
4	4	4	4	1014

Candy crush and Drop

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
110	0	0	0	114
210	0	0	0	214
310	0	0	113	314
410	0	0	213	414
610	211	112	313	614
710	311	412	613	714
810	411	512	713	1014

Stable State

Example Input 2:

[[9,9,7,9,9,9],[7,7,6,8,9,9],[5,6,5,6,8,8],[1,5,1,4,1,1],[2,1,4,1,1,1],[1,4,1,3,1,1],[1,1,2,1,3,1],[1,2,1,1,1,3]]

Example Output 2:

[[0,0,0,0,0,0],[0,0,0,0,0,0],[0,0,0,0,0,0],[0,0,0,0,0,0],[0,0,0,0,0,0],[0,0,0,0,0,0],[0,0,0,0,0,0],[9,9,0,0,9,9]]

Problem 2: Olympic Data Analysis [hw12.py]

Your task is to analyze an olympics dataset (**olympics.csv**), which was derrived from the Wikipedia entry on All Time Olympic Games Medals, and does some basic data cleaning. Use this dataset to answer the questions below. You are required to write some functions, and write some code to execute these functions.

(1) Which country has won the most gold medals in summer games? Write a function named "`answer21()`" that should return a single string value.
The answer is: `United States`

(2) Which country had the biggest difference between their summer and winter gold medal counts? Write a function named "`answer22()`" that should return a single string value.
The answer is: `United States`

(3) Which country has the biggest difference between their summer and winter gold medal counts relative to their total gold medal count? Only include countries that have won at least 1 gold in both summer and winter. Write a function named "`answer23()`" that should return a single string value.

$$\frac{Summer\ Gold - Winter\ Gold}{Total\ Gold}$$

The answer is: `Bulgaria`

Problem 3: Census Data Analysis [hw13.py]

Your task is to analyze a census dataset (**census.csv**) from the United States Census Bureau. Counties are political and geographic subdivisions of states in the United States. This dataset contains population data for counties and states in the US from 2010 to 2015. For a description of the variable names, please refer to **co-est2015-alldata.pdf** .

(1) Which state has the most counties in it? Write a function named "`answer31()`" that should return a single string value. (Hint: consider the sumlevel key carefully! You'll need this for future questions too.)
The answer is: `Texas`

(2) Only looking at the three most populous counties for each state, what are the three most populous states (in order of highest population to lowest population)? Write a function named "`answer32()`" that should return a **list** of string value. (Hint: Use CENSUS2010POP)
The answer is: `['California', 'Texas', 'New York']` if you do not consider SUMLEV
or: `['California', 'Texas', 'Illinois']` if you consider SUMLEV
Note that both answers are correct.

(3) Which county has had the largest change in population within the five year period? Write a function named “`answer33()`” that should return a single string value. (Hint: population values are stored in columns POPESTIMATE2010 through POPESTIMATE2015, you need to consider all six columns.)

e.g. If County Population in the 6 year period is 100, 120, 80, 105, 100, 130, then its largest change in the period would be |130-80| = 50.

The answer is: `Texas` if you do not consider SUMLEV

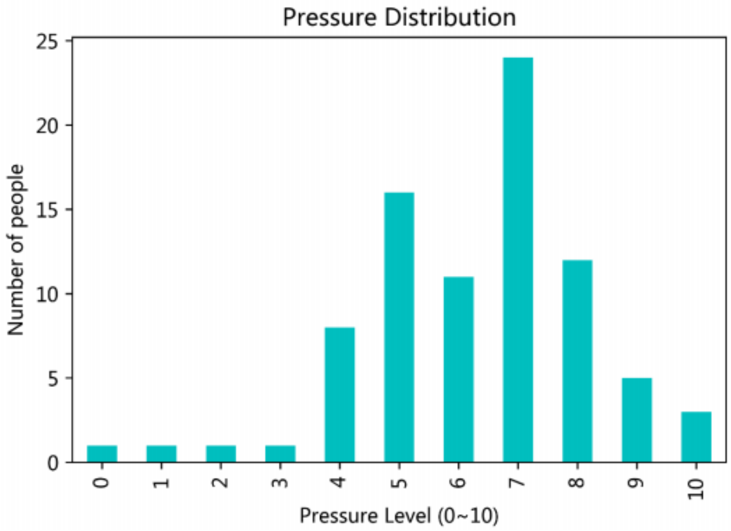
or: `Harris County` if you consider SUMLEV

Note that both answers are correct.

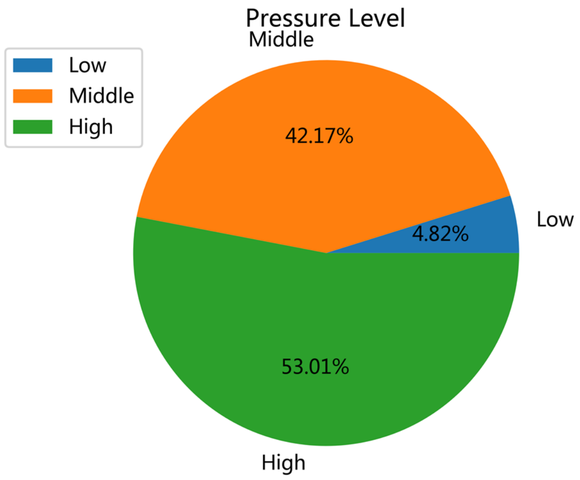
Problem 4: Student Pressure Data Visualization [hw14.ipynb]

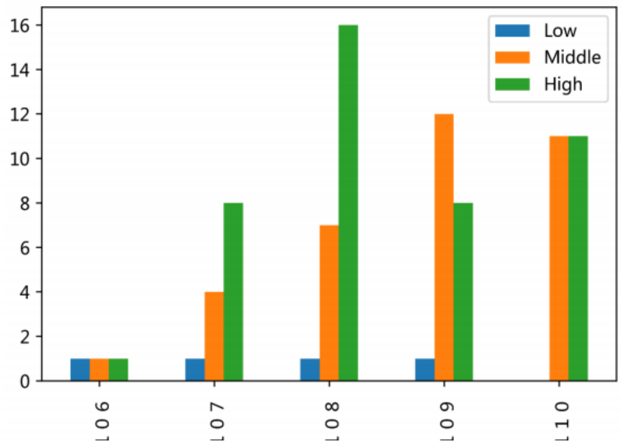
Your task is to analze the relationship between grade (年級) and pressure level (1~10) using the NCKU Statistics student pressure data (**pressure.csv**). Please generate the following results using Jupyter Notebook to write an ipynb named “`hw14.ipynb`”.

(1) The distribution of pressure levels in bar chart (x-axis: pressure levels, y-axis: number of people). You should output the following figure.



(2) The distribution of three pressure categories in pie chart. (Low: 0-3, Middle: 4-6, High: 7-10). You should output the following left figure.





(3) The distribution of pressure categories in different grades. (Grades = 106, 107, 108, 109, 110). You should output the above right figure.

Note that in your codes for these problems,
you need to write some comments to describe the meaning of each part.

How to Submit Your Homework?

Submission in NCKU Moodle. Before submitting your homework, please zip the files (**hw11.py**, **hw12.py**, **hw13.py**, and **hw14.ipynb**) in a zip file, and name the file as "學號_hw1.zip". For example, if your 學號 of your team are H12345678, then your file name is:

"H12345678_hw**1**.zip" or "H12345678_hw**1**.rar"

When you zip your files, please follow the instructions provided by TA's slides to submit your file using NCKU Moodle platform http://moodle.ncku.edu.tw .

Have Questions about This Homework?

Please feel free to visit TAs, and ask/discuss any questions in their office hours. We will be more than happy to help you.